2008-01-0776

Virtual Combat Vehicle Experimentation for Duty Cycle Measurement

Mark Brudnak, Mike Pozolo, AnnMarie Meldrum
U.S. Army RDECOM-TARDEC

Todd Mortsfield, Andrey Shvartsman
DCS Corporation

Wilford Smith, Jarrett Goodell
SAIC, Inc.

Dale Holtz
Realtime Technologies, Inc.

ABSTRACT

This paper describes a human-in-the-loop motion-based simulator which was designed, built and used to measure the duty cycle of a combat vehicle in a virtual simulation environment. The simulation environment integrates two advanced crewstations which implement both a driver's station and a gunner's station of a simulated future tank. The simulated systems of the tank include a series hybrid-electric propulsion system and its main weapon systems. The simulated vehicle was placed in a virtual combat scenario which was then executed by the participating Soldiers. The duty cycle as measured includes the commands of the driver and gunner as well as external factors such as terrain and enemy contact. After introducing the project, the paper describes the simulation environment which was assembled to run the experiment. It emphasizes the design of the experiment as well as the approach, challenges and issues involved. It presents the experiment results and briefly discusses on-going and future work.

INTRODUCTION

One of the goals of the RDECOM-TARDEC Power and Energy (P&E) program is to advance the design, development, and testing of hybrid electric power and propulsion technology for advanced combat vehicles. This is being accomplished through the integration and evaluation of power and energy technologies from various Army Technology and Objective (ATO) programs. The by-product of the TARDEC P&E program will be a compact, integrated system that will provide efficient power and energy generation, and power management, suitable for spiral integration into the Future Combat System (FCS) Manned Ground Vehicle (MGV) program.

To effectively develop an advanced power system for combat vehicles, accurate estimates of power loads throughout the complete range of operations are required. A comprehensive combat vehicle usage profile, or "duty cycle", which would provide these power load estimates, does not exist at this time. The TARDEC P&E program is attempting to remedy this situation by establishing multiple combat vehicle duty cycles. These duty cycles will be derived from the virtual representations of advanced combat vehicles and combat scenarios using both warfighter-in-the-loop and power system hardware-in-the-loop simulation described in detail in the remainder of this paper. These duty cycle measurements combine engineering-level power supply systems with performance-level models of power consumption devices within a warfighter simulation of combat mission scenarios.

For our purposes, a military vehicle's *duty cycle* is specific to the mission and platform type but is a design- and configuration-independent representation of events and circumstances which affect power consumption. Such events and circumstances encompass (1) vehicle operation such as speed, grade, turning, turret/gun activity, and gun firing plus (2) external scenario components that affect power consumption like incoming rounds, ambient temperature, and soil conditions. The event inputs can be distance-based when the vehicle is moving or time-based when the vehicle is stationary, or triggered with some other state condition.

UNCLAS: Dist A. Approved for public release.

Report Documentation Page

Form Approved
OMB No. 0704-0188

1. REPORT DATE **18 JAN 2008**	2. REPORT TYPE **N/A**	3. DATES COVERED **-**
4. TITLE AND SUBTITLE **Virtual Combat Vehicle Experimentation for Duty Cycle Measurement**		5a. CONTRACT NUMBER
		5b. GRANT NUMBER
		5c. PROGRAM ELEMENT NUMBER
6. AUTHOR(S) **Mark Brudnak; Mike Pozolo; Ann Marie Meldrum; Todd Mortsfield; Andrey Shvartsman; Wilford Smith; Jarrett Goodell; Dale Holtz**		5d. PROJECT NUMBER
		5e. TASK NUMBER
		5f. WORK UNIT NUMBER
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) **US Army RDECOM-TARDEC 6501 E 11 Mile Rd Warren, MI 48397-5000**		8. PERFORMING ORGANIZATION REPORT NUMBER **18609RC**
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)		10. SPONSOR/MONITOR’S ACRONYM(S) **TACOM/TARDEC**
		11. SPONSOR/MONITOR’S REPORT NUMBER(S) **18609RC**

12. DISTRIBUTION/AVAILABILITY STATEMENT
Approved for public release, distribution unlimited

13. SUPPLEMENTARY NOTES
Presented at the SAE 2008 World Congress, April 14-17, 2008, Detroit, Michigan, USA, The original document contains color images.

14. ABSTRACT

15. SUBJECT TERMS

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT **unclassified**	b. ABSTRACT **unclassified**	c. THIS PAGE **unclassified**	**SAR**	**17**	

To measure such a duty cycle, the TARDEC Simulation Laboratory (TSL) has been building a motion base/warfighter-in-the-loop simulation capability in which Soldiers can virtually operate their vehicles in relevant combat scenarios. This capability is then used to perform experiments in which duty cycle information is captured. These experiments are called duty cycle experiments (DCEs).

TARDEC has thus far completed three DCEs. The first, executed in November 2005, was called DCE1. It consisted of a single vehicle with a driver on a motion base simulator. It is more fully described in [1,7]. The second experiment, conducted in June-July 2006, was called DCE2. This experiment simulated a single FCS Mounted Combat System (MCS) vehicle and incorporated both a driver and a gunner. The experiment is described in [2,3,6]. The latest experiment conducted in May-June 2007 was called DCE3. This experiment simulated two MCS vehicles, one under motion and one static, both with a driver and a gunner. The DCE3 experiment is described in the remainder of the paper. The three DCE experiments are compared and contrasted in Table 1.

Table 1. Comparison of the three DCEs.

	DCE1	DCE2	DCE3
Date	Nov ‘05	June ‘06	May ‘07
Participants	Civilian	Military	Military
Runs	7	12	12
Scenarios	1	1	2
Vehicles	1	1	2
Roles	Drive	Drive/Gun	Drive/Gun
Motion base	RMS	RMS	TMBS
Length	11 km	13 km	61/38 km
Duration	25 min	25 min	100/40 min
Long Haul		Yes	Yes
BLOS			Yes
NV/IR			Yes
Moving troops			Yes
Wingman			Yes

Each of the three DCE experiments was designed to measure an accurate duty cycle given the available resources and technology. As a matter of standard practice, each participant was asked to comment on the quality and realism of the simulation. This feedback was used to improve the subsequent experiments. For DCE3 the objectives centered on executing a relevant scenario, providing a wingman to the primary vehicle, and providing infra-red (IR) and night vision (NV) capability to the crews. Each of these objectives was attained in the DCE3 experiment. A particularly novel aspect of the DCE2 and DCE3 experiments was the hardware-in-the-loop (HWIL) integration of the P&E Systems Integration Lab (SIL) to function as the power train of the primary vehicle. This aspect of the simulation is referred to as the *long haul* or *RemoteLink* component in the remainder of the paper.

As mentioned earlier, the fundamental objective of the DCE3 experiment was to measure the duty cycle of an MCS vehicle in a relevant scenario. Such a duty cycle consists of participant behavior (as represented by their use of vehicle controls) and external circumstances (enemy, ground, lighting, weather). For this participant behavior to be realistic, the simulation environment must be as accurate as possible. We therefore invested a lot of time and effort in the fidelity of the scenario, tasks and virtual environment. For the scenario we derived two independent variants, one provided by Ft. Knox and one based on CASTFOREM both implemented in OneSAF Test Bed (OTB). The tasks of driving and gunning were accomplished via controls and displays provided by an advanced prototype crewstation called the Crew-integration and Automation Test-bed (CAT) crewstation. The virtual environment was rendered as motion, visuals and sound. The motion was provided by the Turret Motion Base Simulator (TMBS); the visuals were provided by the Night Vision Image Generator (NVIG); the sound was generated by a product called SimCreator®, which also functioned as our integration framework. Furthermore, to integrate the P&E SIL, we developed a custom solution for the robust integration of the SIL into the real-time environment.

In the remainder of the paper we discuss the experiment design with regard to the scenarios and run sequences. We then discuss the simulation architecture and design to include a description of the major components. We discuss the implementation of the long haul connection to the P&E SIL. Finally, we present some representative results and offer some conclusions.

EXPERIMENT DESIGN

As mentioned, the DCE3 experiment used two independent combat scenarios. The reasons for this are two-fold. First, to make the fullest use of our participants’ time, we wanted them to spend the most time “on simulator” as possible. Second, we received candidate scenarios from two different sources both of which had appealing features. They were markedly different so we chose to use both. The first and what we called primary scenario was developed by the Unit of Action Maneuver Battle Lab (UAMBL) at Ft. Knox, KY. The second scenario was derived from an execution of the Combined Arms and Support Task Force Evaluation Model (CASTFOREM). Both scenarios employ the MCS’s main and auxiliary weapon systems in line-of-sight (LOS) engagements and employ what are called beyond-line-of-sight (BLOS) engagements. In BLOS engagements, targets may be engaged which are not in direct view. The UAMBL scenario emphasized LOS engagements and the CASTFOREM scenario emphasized BLOS engagements. Furthermore, both scenarios are executed on Ft. Knox terrain using the same route. These scenarios are briefly described in the following sections.

UAMBL SCENARIO

The scenario delivered by UAMBL was written for a company-sized element consisting of one (Mounted Combat System) MCS platoon and two Infantry Carrier Vehicle (ICV) platoons whose mission is to enter hostile territory, assault an objective, and rapidly return. It is set at night so all driving/gunning occur through NV/IR displays. It is specifically written for the MCS platoon as the simulated entity and occurs in three phases: phase 1 consists of a rapid advance to an objective, phase 2 consists of a support by fire position at the objective, and phase 3 consists of the exit operation to return to friendly

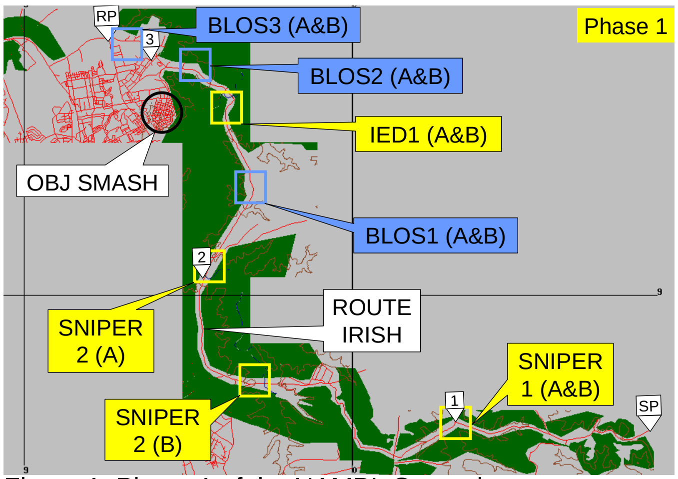

Figure 1. Phase 1 of the UAMBL Scenario.

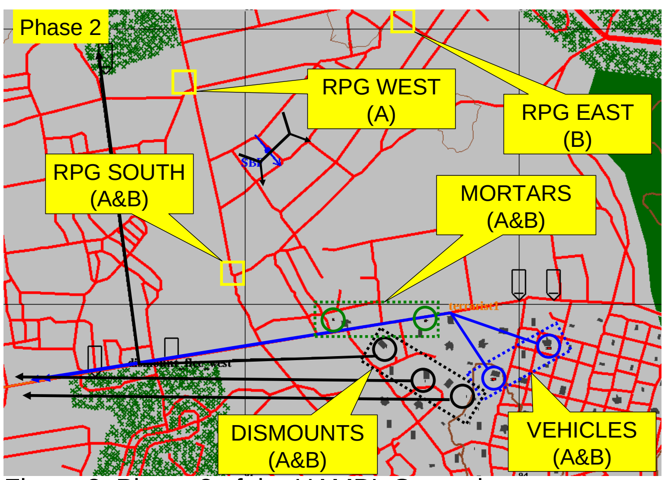

Figure 2. Phase 2 of the UAMBL Scenario.

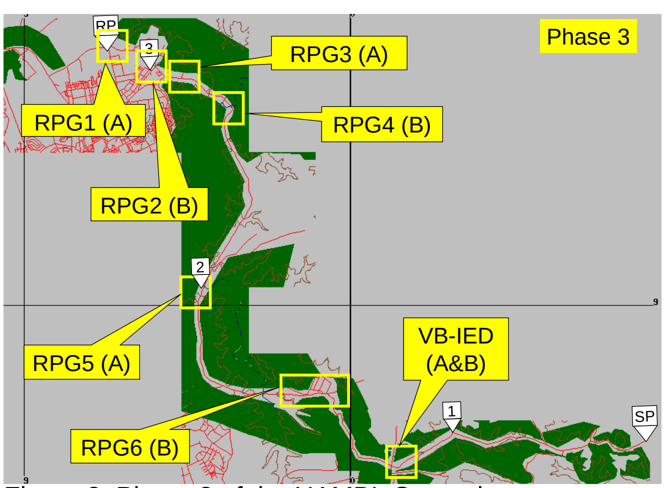

Figure 3. Phase 3 of the UAMBL Scenario.

territory. We elected to implement all three phases of the scenario for our simulation.

In phase I, the engagements occur on the route IRISH which may be seen in Figure 1. The goal for phase 1 is to reach the objective as soon as possible. The platoon encounters a few enemy ambushes on the route and an improvised explosive device (IED). They preemptively engage the objective with BLOS while en-route. Phase 2 begins once the support by fire (SBF) is set (see Figure 2). During the SBF the platoon continues to engage the enemy in relation to the objective. Engagements consist of vehicles, dismounts, mortar teams and RPGs. In phase 3 the company reconstitutes its column and returns along route IRISH (see Figure 3). The enemy, organizes ambushes along the exit route with RPGs. The enemy also positions a Vehicle Borne IED (VB-IED) for the returning company. The company may address the VB-IED with a BLOS engagement or a LOS engagement.

CASTFOREM SCENARIO

The CASTFOREM scenario was developed for DCE3 using a scenario situated on a different terrain. The scenario described how an FCS brigade rapidly moves from a staging area to a position in close contact with enemy forces in preparation for an assault on an enemy stronghold. The scenario incorporates the rapid advance and positioning of forces as well as the engagements with perimeter forces along the way. To simplify the implementation, the scenario was transposed from its original terrain onto the Ft. Knox terrain. Unlike the UAMBL scenario, this scenario traversed the route in only one direction. The engagements which occur along the route are shown in Figure 4. As may be seen, the engagements are mostly BLOS. The few LOS engagements which occur are against 2-3 dismounts.

EXPERIMENT EXECUTION

Twelve Soldiers participated in the DCE3 experiment; four Soldiers per week for three weeks. Additionally, a senior NCO (E7/E8) participated for all three weeks as

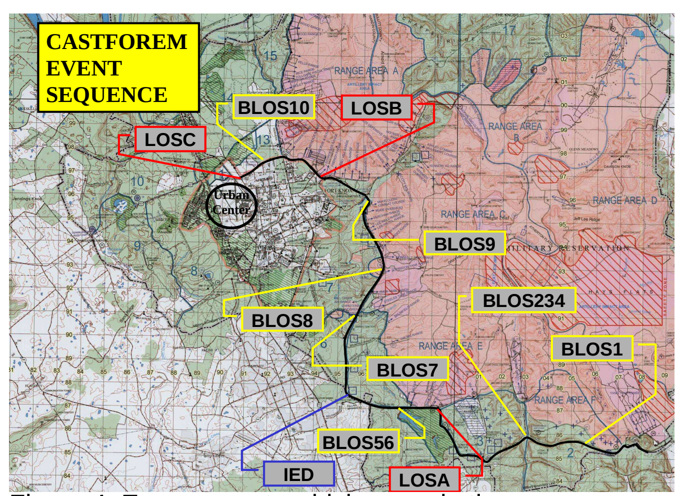

Figure 4. Engagements which occur in the CASTFOREM scenario.

Figure 5. (left) TMBS with crewstation platform, enclosure and two CAT crewstations attached and (right) two CAT crewstations on the crewstation platform.

Figure 6. Secondary (wingman) crewstations. (left) Secondary gunner's station consisting of one touch screen display and a joystick controller. (right) Secondary driver's station consisting of a day-view display, a steering wheel, and two pedals.

the platoon commander. This NCO was referred to as the "proxy commander." The four participating Soldiers were divided up into two crews each consisting of a driver and a gunner. One crew served as the primary crew and the other served as the secondary crew. The primary crew operated the vehicle on the motion base while the secondary crew operated the stationary vehicle. The motion base along with the primary crew interface may be seen in Figure 5. The secondary crewstation may be seen in Figure 6.

By design, each crew was given an opportunity to function as both the primary and secondary crew each experiment day. For one run they would be the primary and next the secondary. Furthermore, each crew would maintain their assigned roles (i.e., driver and gunner) for the first two experiment days (Monday and Tuesday); for the next two days (Wednesday and Thursday) they would swap roles. This scheduling scheme is illustrated in Figure 7. There the Soldiers are labeled S01 through S04. The rows indicate their assignment to teams and roles; the columns indicate the time sequence of the four configurations shown. On Tuesday, for example, the crews would execute the "A" version of the scenario. They would then switch from primary to secondary and execute the "B" version of the scenario. The A and B versions of the scenario differed in red force positioning. Two scenarios were necessary to prevent the gunners from memorizing the scenario.

The Crew Station / Turret Motion Base Simulator (CS/TMBS) was used for all tests and operated by TARDEC engineers. The tests were conducted from 21 May – 14 June 2007, with 12 Soldiers as participants from Ft. Bliss, Texas. Each participant was given an initial overview/brief of the test objective and a safety

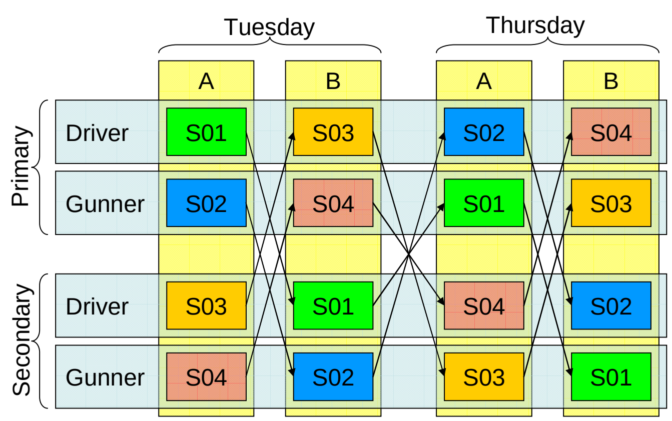

Figure 7. Configuration of four soldiers into teams and roles throughout the experiment week.

brief of the simulator before participating in the actual experiment. The participants filled out a "Demographics and Personal Experience Questionnaire" which gathered information on their background and experience. The experiment procedures were the same each test day. They consisted of experiment preparations each morning that included warming up and exercising the CS/TMBS, prepping the software by configuring and starting up the software programs for the scenarios, terrains, tactical map, and powering up the CAT crewstations. To execute each experiment, the participants boarded the CS/TMBS and were seated and belted in the CAT crewstations. Each was administered a "Motion Sickness Questionnaire" prior to and following each experiment so that the operator could gauge whether the occupants were feeling any motion sickness. Upon completion of the experiment, the subjects were administered qualitative surveys and questionnaires regarding their experience and opinions on the experiment.

SIMULATION ARCHITECTURE AND DESIGN

The DCE3 experiment was comprised of several independent systems that were integrated to provide the functionally necessary to support two operators, each controlling a crewstation cockpit on a 6-DOF motion platform in an immersive synthetic battlefield environment. In this section we discuss the design and architecture and of the simulator which was used to execute the experiment.

The major components of the DCE3 simulator are the TMBS, crewstation enclosure, the CAT crewstations, the secondary crewstations, the ESS rack and the set of computers used to run the simulator. There were two CAT crewstations mounted on the TMBS (shown in Figure 5) for the primary crew. The secondary crewstations, were stationary and located in the laboratory.

The simulator that was built to conduct the DCE3 experiment consisted of 30 Intel®-based PCs inter-networked with five independent 100 MBPS switched Ethernet subnetworks. The interconnections and placements of these components in the system are illustrated in the wiring diagram shown in Figure 8. In this figure, the location *TMBS* denotes those components on the motion platform, the *AREA 5&6* denotes the lab area containing the TMBS, the *CONTROL ROOM* denotes the area where the

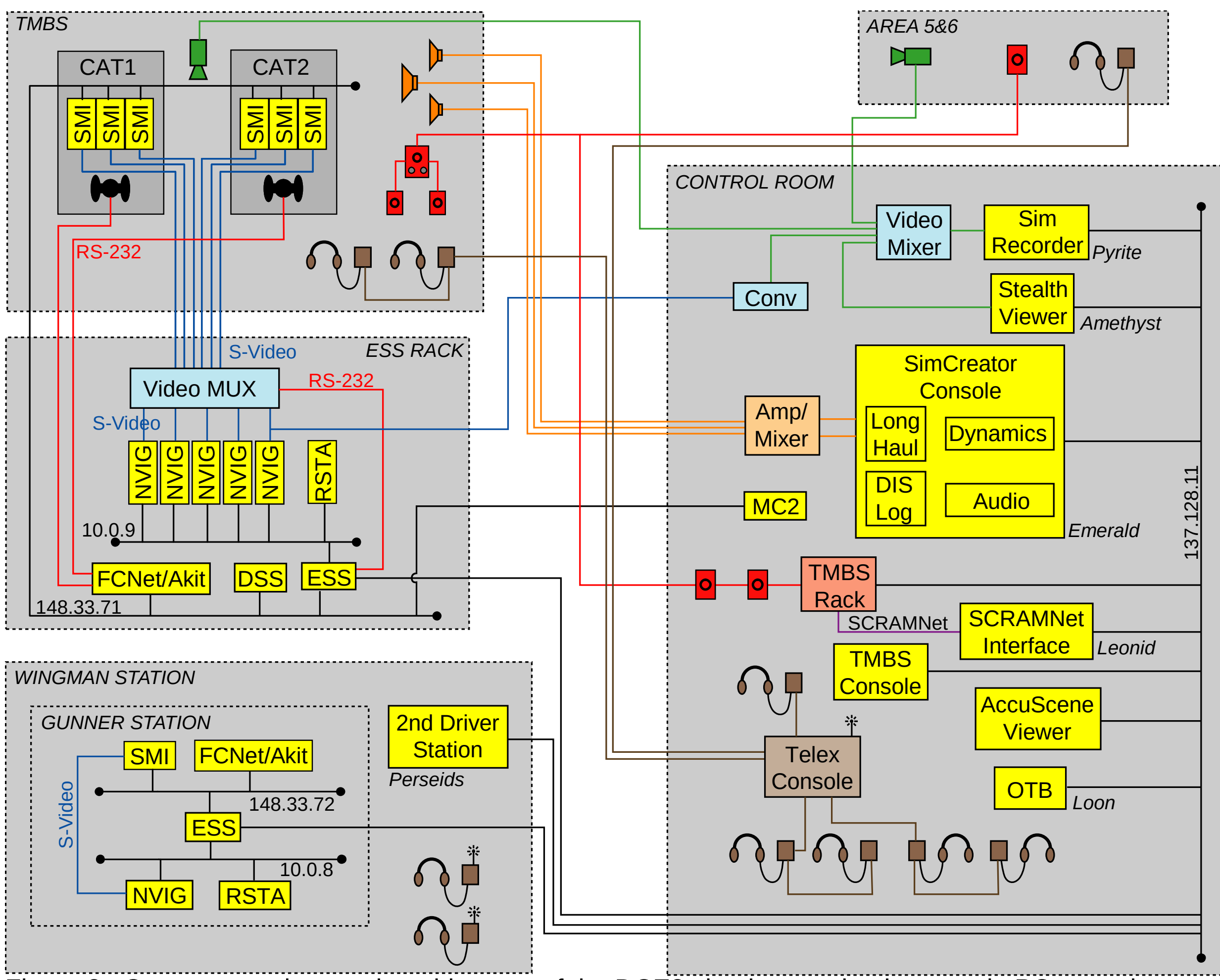

Figure 8. Computer and network architecture of the DCE3 simulator as implemented. PCs are shown as yellow boxes. Location-based groupings are indicated with grey boxes.

simulation operators were positioned, the *ESS RACK* is the 19" rack also located in the control room and the *WINGMAN STATION* is located in the control room as well. The primary means of communication among these computers is the TSL local area subnetwork (137.128.11.*) shown on the right. Also shown are the signal flows associated with the CAT crewstations (which have dedicated subnetworks), the audio subsystem, the video monitoring and recording subsystem, the communication system, the E-stop system and the motion control system.

As can be seen the simulator was comprised of a variety of existing subsystems all integrated to form the whole. The integration was mostly accomplished by means of SimCreator, which is a commercial product developed by Realtime Technologies, Inc. (RTI) [8]. It was used to seamlessly integrate the main processes on *Emerald*, *Leonid*, *Amethyst*, and *Pyrite*. *Emerald*, a quad-core Windows XP® PC, lies at the heart of the simulation architecture. It executes one process per core; these processes are vehicle dynamics, the long haul interface, the DIS logger and the audio process. *Leonid* served as an interface between SimCreator and the TMBS by placing motion commands on the SCRAMNet® interface. *Amethyst* ran the stealth viewer which gives a "parasail" view of the primary vehicle (using SimCreator's IG). *Pyrite* runs sim-recorder which records four video channels and one audio channel in MPEG format.

Several other computers were not integrated using SimCreator. *Loon* is a Linux PC running OneSAF Test Bed (OTB) which is responsible for generating the red forces. The AccuScene Viewer generates a stealth view of the battle rendering all entities on the DIS network. The TMBS console is responsible for controlling the VME controllers in the TMBS Rack; it is the sole interface to the TMBS. The MC2 laptop was used to "drop" BLOS targets into the simulation environment. The primary CAT crewstations exclusively communicate with the simulation via the ESS box, thus abstracting their internal complexity away from the simulation architecture. The secondary (wingman) station is abstracted in a similar

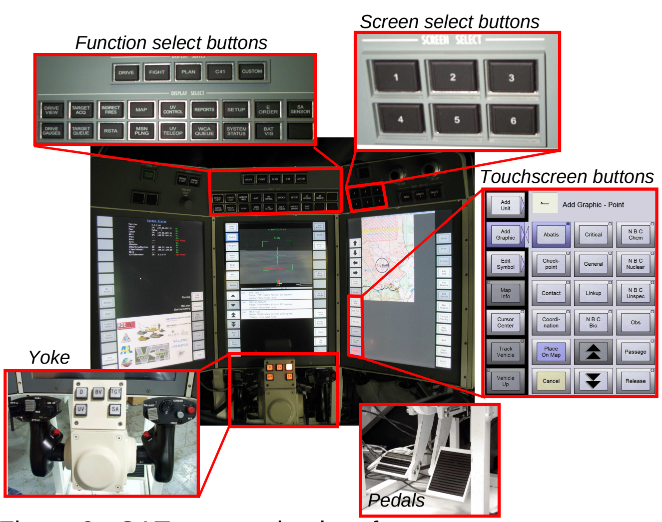

Figure 9. CAT crewstation interfaces.

sense. The secondary driver station (*Perseids*) is a stand alone SimCreator process which implements the secondary vehicle dynamics and communicates with the secondary ESS directly. The primary and secondary vehicle stations are completely independent and communicate using only the DIS protocol.

The primary driver's and gunner's interfaces were provided by the CAT crewstation (see Figure 9). The CAT crewstation is a stand-alone simulator used to evaluate operational effectiveness of a two-man crew for future combat vehicles. The crewstation consists of three 43cm x 33cm (17" x 13") touch screen panels, several dedicated pushbuttons, a yoke, and foot pedals. The operator interface on the crewstations is controlled by the Soldier-Machine Interface (SMI) computers which communicate with the Embedded Simulation System (ESS) over a dedicated Ethernet subnet (TCP/IP and UDP). Video is provided to the CATs by up to three Image Generator (IG) computers via a standard S-Video interface. The IG computers generate their scenes using the Night Vision Image Generator (NVIG) developed by Nigh Vision Labs (NVL) of the U.S. Army Communications and Electronics Research Development and Engineering Center (CERDEC). The IG channels are directed to the proper SMI display by means of a video multiplexer which is controlled by the ESS. The RSTA server is responsible for performing line intersection queries for ballistics and laser range finding. The FCNet process is responsible for modeling the vehicle's weapons. The A-kit is responsible for abstracting the ESS from the rest of the system. The DSS computer is responsible for generating the target lists as specified by the MC2 laptop.

The Desktop CAT is a PC-only implementation of the CAT crewstation, which does not require the crewstation hardware to run. One such station was used to implement the secondary gunner's station. It consisted of 1 SMI channel, 2 NVIG channels (1 IG and 1 RSTA server), the Akit/FCNet computer and the ESS computer. Each of these serves the identical role they do in the primary station. The Desktop ESS had to be modified to allow for a gunner-only operation mode, with all of the driver actuation commands coming from SimCreator.

SimCreator was used to model the FCS-like vehicle dynamics and the turret/gun system mounted on the vehicle. SimCreator communicated with the crewstation through the Embedded Simulation System (ESS). ESS and SimCreator used a common socket-based communications protocol called OE, which was designed to emulate shared memory across a network. It provides facilities for creating message stores for continuous data such as driver and gunner actuators and vehicle state, as well as message queues for passing discrete events such as gun fires and EM armor discharges. Unlike previous experiments, the ESS was modified to so that all weapon modeling and vehicle dynamics was handled by SimCreator. In this experiment, the ESS acted as a message distribution system between all the components in the simulation. SimCreator passed

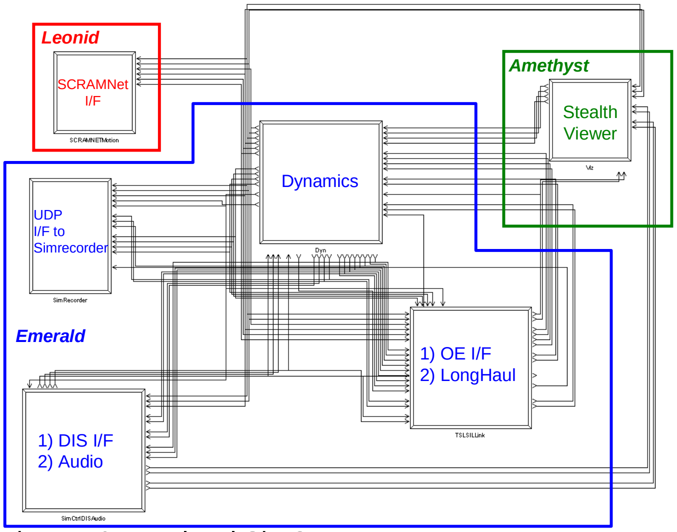

Figure 10. Top-level SimCreator components.

vehicle and turret motion data to the motion base controller through the SCRAMNet shared memory at 100 Hz.

The SimCreator portion of the simulator was implemented as a distributed simulation consisting of three computers communicating on the same subnet via UDP and FTP protocols (see Figure 10). When the simulation is started, the master SimCreator process (on Emerald) connects to the other processes and distributes the initial conditions via FTP and starts the distributed processes using REXEC. Once the simulation is running, all of the shared data between the distributed processes is passed back and forth via the UDP protocol. Each of the SimCreator computers serves a pivotal role in the simulation. The master computer (*Emerald*) runs the vehicle dynamics, interfaces to the ESS via the OE, communicates with the SIL via LongHaul interface, interfaces to the DIS network, runs the DIS logger, generates the audio, controls video recording via the sim-recorder interface, and provides a control GUI. This master computer is quad-core machine.

The Visuals computer (*Amethyst*) provides a local stealth view of the primary vehicle for the experiment operators. This stealth view also generates one of the four views recorded in the sim-recorder software. The SCRAMNet interface computer (*Leonid*) passes vehicle state information to the SCRAMNet ring to be read by the TMBS controller.

COMPONENT DESCRIPTIONS

As stated earlier the DCE3 experiment consisted of several independent components which were integrated together to form a complete simulator. This section describes some of the major hardware and software components used in the simulator.


Figure 11. Turret Motion Base Simulator (TMBS) with Reconfigurable Platform.

Table 2. Turret Motion Base Simulator (TMBS) performance specifications.

Translational Motion	
Displacement	±30 in (76 cm)
Velocity	±70 in/s (178 cm/s)
Acceleration (*Max ind. transient)*	±6 g
Bandwidth) (*3 dB Frequency)*	10 Hz
Rotational Motion	
Displacement	±20 deg
Velocity	±70 deg/s
Acceleration	±2922 deg/s^2
Max Payload	50,000 lbs (22,680 kg)

TURRET MOTION BASE SIMULATOR (TMBS)

The TMBS (Figure 11) was used as the motion rendering device for all tests. It is a high-capacity, 6 degree of freedom (DOF) motion simulator, which can accommodate both re-configurable crewstations and active turret systems. It is comprised of a platform mounted on six symmetrically positioned hydraulic actuators. It produces motions in the longitudinal, lateral, vertical, roll, pitch, and yaw directions (see Table 2). The simulator can accommodate turret systems as heavy as an M1A2 Abrams class turret (approximately 25 tons). The TMBS has the capability of using an active, fully functional, field-ready turret or crewstation during a manned simulation. With a crew controlling all turret or crewstation functions, the TMBS can replicate the dynamic disturbances the entire hull would generate and

experience as if the complete vehicle system was traveling over various rough cross country terrains. The simulator possesses a frequency bandwidth of approximately 10 Hz, making it possible to accurately replicate cross-country disturbances to the crew. (Such motions are crucial to the accuracy, realism and validity of simulated operations over rough terrain.)

The simulator has been safety certified for use by Soldiers and experimenters in accordance with the Army Regulation (AR) 70-25. TARDEC conducted a safety assessment of the simulator for man-rating purposes. Furthermore, the RDECOM safety office as well as the U.S. Army Installation Management Agency (IMA) - Installation Safety Office evaluated the safety of the TMBS; they have approved the TMBS for human-in-the-loop experiments. The simulator is equipped with a safety interlock system to ensure the ride motion does not exceed safe positions or accelerations.

For the DCE3 Experiment, a crewstation platform (black truss and silver rails in Figure 11) was mounted to the TMBS allowing two CAT crewstations to be affixed to it. The platform provides a large surface on which to mount experimental hardware such as the CAT crewstations. The two CAT crewstations were covered with black canvas to occlude stray ambient light from the operators.

CAT CREWSTATION

The Crew Integration and Automation Testbed (CAT) crewstation is a TARDEC development effort representing several iterations of research and development stemming from the early generation Crewman's Associate, Vetronics Technology Testbed (VTT), and the Demo III Unmanned Ground Vehicle (UGV) Tasking programs. The product of extensive research in the area of Human Factors Engineering, Usability and Systems Engineering, the CAT crewstations incorporate many years of research and user feedback from various experiments conducted throughout the program.

The CAT crewstation consists of three vertically mounted touch screen Multi Function Displays (MFDs) as the primary visual interface, hard button function selection buttons (located above the center display), and the pedal/yoke interface (see Figure 9). Each of the vertical MFDs are divided into two virtual screens, thereby providing a total of six virtual screens. These screens allow the operator to conduct any of the various functions: target acquisition and engagement, indirect vision driving and teleoperation of robotic assets, command and control, navigation and mission planning, battlefield visualization, and embedded training and mission rehearsal.

The Soldier Machine Interface (SMI) software links the various multimodal interfaces of the CAT and allows the operator a high degree of personalization in terms of screen layout. For example, the Map and Mission Planning screens may be placed on any of the six virtual screens. Similarly, the teleoperation and target acquisition screens may be placed in any of the top three screens (1,2,3) per the operator's preference. Upon activation of a function screen, the touch screen buttons pertinent to each function appear on the selected screen. The technical architecture of the SMI software allows extension of screens, thereby allowing integration and development of additional components such as survivability suites.

For the DCE3 experiment two identical CAT crewstations were each configured to operate in either the drive function or the target acquisition and engagement (gunner) function.

POWER SYSTEM MODEL

Hybrid power systems take on a number of different configurations broadly classified as either series or parallel architectures. Within these classes, there exists a myriad of possible configurations, topologies and component alternatives. Any tool that purports to analyze hybrid drive alternatives must be flexible enough to allow the user the ability to construct all of these electrical/mechanical configurations. The P&E program has developed a software library of Simulink® components (called CHPSPerf) consisting of basic mechanical and electrical components such as conventional gears, planetary gears, and electrical machines. These are assembled within the graphical environment to rapidly build models of arbitrary power systems. Interactions between these components are described by the graphical connections that mimic the physical connections between the actual components. This means the models can be intuitively organized in a manner similar to the physical construction of the system.

Power systems that have been simulated with the tools contained in the toolbox include purely electrical power systems as well as purely mechanical power systems and the various hybrid systems. As a concrete example, consider the series-hybrid design shown in Figure 12 (which also happens to be the P&E SIL topology and a variant of the FCS MGV topology). This system uses a

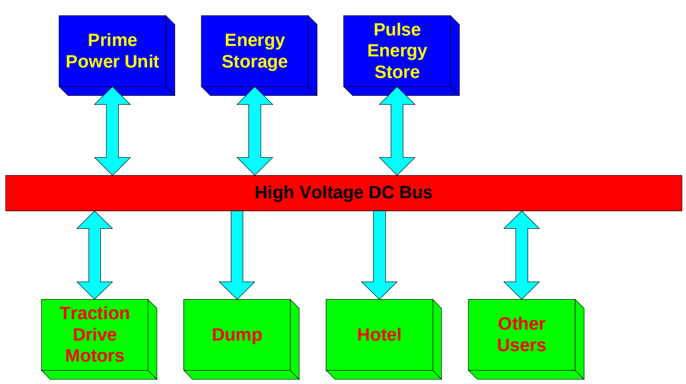

Figure 12. Layout and components of the series hybrid power system used in the CHPSPerf model.

diesel engine coupled to an induction motor/generator unit (Prime Power in Figure 12) to provide continuous power through an inverter to an unregulated high voltage DC bus. A battery pack (Energy Storage in Figure 12) sized to provide silent watch and silent mobility functions is attached directly to the bus and maintains bus voltage at approximately 600 Volts. Attached to the high voltage bus are two independent induction motors for the left and right sprocket drives (Traction Drive Motors) capable of providing 300 kW of continuous power and over 900 kW of burst power for braking and acceleration functions. A steer motor was used to develop differential torque for high speed steering. A brake or dump resistor is also attached to the bus to protect it from over-voltage conditions that might arise due to heavy braking or long duration regeneration events.

For the purpose of interfacing to the GVSL vehicle model in the SimCreator environment, a static link library was created from the Simulink block diagram shown in Figure 13. The Real-Time Workshop® (RTW) of the Simulink environment was used to generate the C code used to create the library. Simulink S-functions were used to define the calling interfaces to the power system library routines and SimCreator. This was done to encapsulate the internal implementation details of the power system model. In this way changes to the power system model could be done transparently and robustly.

Each of the components shown in the notional layout of Figure 12 is modeled in the CHPSPerf series hybrid simulation, shown in Figure 13. The subsystems of primary interest to DCE3 are contained in the blocks labeled as High Voltage Power Train/Energy Storage and ILR Energy Management Controller in Figure 13. A summary description of the underlying models is given below.

Vehicle – For the DCE3 experiments vehicle mobility loads are imposed using the multi-body model of the vehicle chassis and suspension. The SimCreator vehicle model effectively wraps the power system model code (generated by RTW) so that it behaves like a SimCreator component. The power system interface, shown in Figure 13 as the two blocks *simCreatorDataIn* and *simCreatorDataOut*, consists of a specification of the direction of the torques and information (torque/speed) flow between the vehicle model and the power system model. In the current interface the power system passes torque information over to the vehicle system and the vehicle system passes shaft speed information back across to the power system.

Motor/Generator – The vehicle uses 3-phase induction machines for the traction motors, steer motor, and the generator. Additionally, the cooling fan is also an induction machine. Because of the relative importance of the mobility system in the overall power system efficiency (accounting for upwards of 90 percent of the total energy consumption during a typical mission) we have expended a substantial effort in developing reliable and accurate machine models for this aspect of the system. The model used in DCE3 is an electrically-steady mechanically-dynamic induction machine model. In this model the machine is approximated electrically as a lumped parameter *LR* circuit in dq-space, i.e., the 3-phase machine is reduced to the equivalent 2-phase machine whose lumped parameter circuit is solved for currents and electrical torques given the terminal voltages. The electrical torques are used in conjunction with the machine inertia and frictional losses in the bearings to find the machine's rotor speed as a function of time, load and bus voltage.

Battery – The Li-ion battery model is represented by a capacitor/resistor network with the values of the various

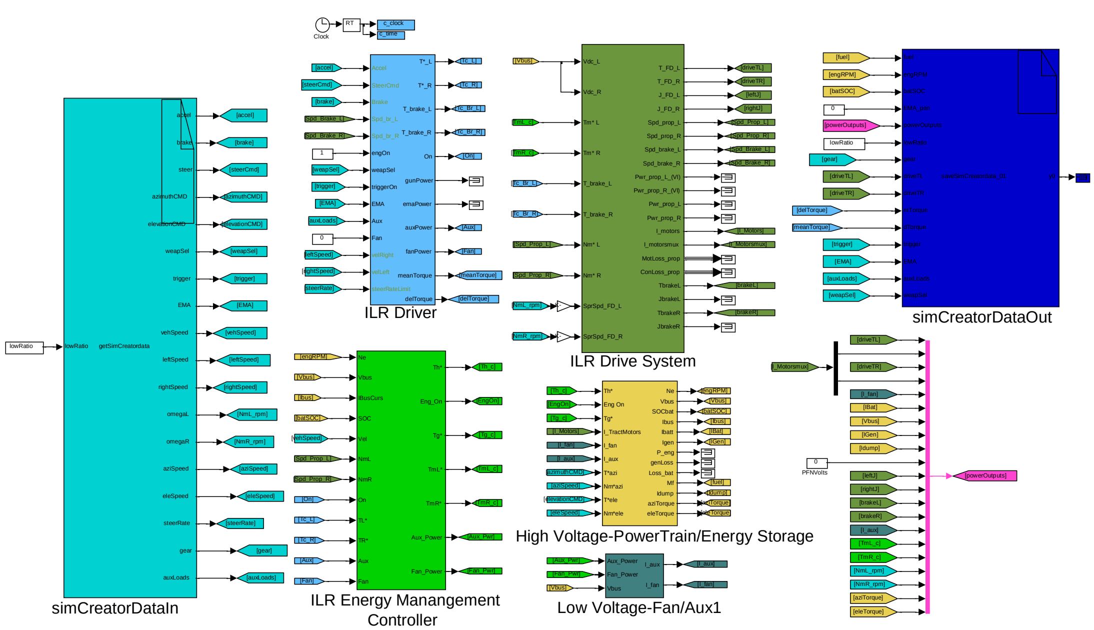

Figure 13. Top-level Simulink model of CHSPerf showing S-function interface to external programs (simCreatorDataIn and simCreatorDataOut).

elements in the system implemented as functions of temperature and state of charge of the battery pack. The parameters are derived from experimental measurements. The single cell model is extrapolated to multiple cells by using appropriate current/voltage transforms for multiple series/parallel combinations of cells.

Engine – The engine model is based on a simple table lookup of the torque and fuel consumption properties. The engine includes no dynamics and is modeled purely as a table look-up. Two tables required for the model are:

Torque table – a two-dimensional table with torque as a function of 'throttle' position (actually for a diesel engine the fuel rail position) and engine speed, and

Specific fuel consumption table – a two dimensional table with SFC as a function of 'throttle' position and engine speed.

Dump Resistor – The dump resistor is modeled as a resistor with a resistance that varies from zero to its maximum value with a linear gain.

Converter – The converter model is based on a loss model that accounts for both passive component (capacitor) and active switching losses. Calculation of the passive losses is performed using the equivalent series resistance of the capacitor of the system. The active losses are calculated using the diode and switch losses during turn-on, turn-off and steady-state standoff. The losses for the system can be put into a form per switch/diode pair.

DYNAMICS MODEL

The vehicle modeled for this experiment is a 27 ton tracked combat vehicle. The tracks have a front drive sprocket, rear idler and 6 road wheels with a trailing arm suspension supported by a torsion bar. The vehicle also has an unmanned turret with a gun.

The vehicle was modeled using SimCreator's vehicle modeling and multibody dynamics components [9]. The chassis, road arms, gun and turret are modeled as individual bodies using a relative coordinate formulation. A picture of the over-all dynamics model may be observed in Figure 14.

The track was modeled as a simple elastic band. Tensions from the track are applied to the sprocket, idler and road wheels. The track-terrain interaction was modeled using Bekker's pressure-sinkage equation [13] to get the normal force and the longitudinal motion resistance due to ground contact pressure. A combined shear displacement, based on the lateral and longitudinal slip ratios, was used to determine the tractive and lateral forces [13]. A schematic of the track model is found in Figure 15.

The speed of the track is determined by the rotation of the drive sprocket. The sprocket dynamics includes the torque from the powertrain, the track tension forces and a resistance torque that models the internal friction of the track. This internal friction is a major contributor to the sprocket dynamics and is essential for accurate duty cycle measurements.

The gun elevation and turret azimuth are controlled by the powertrain model. A small damping force is included in the turret rotation and bump stops are used to limit the gun elevation. Also included are forces to model the gun recoil forces.

LONG HAUL DESIGN

The goal of the *long haul* or *RemoteLink* is to provide a real-time cross-country link that causes TARDEC's

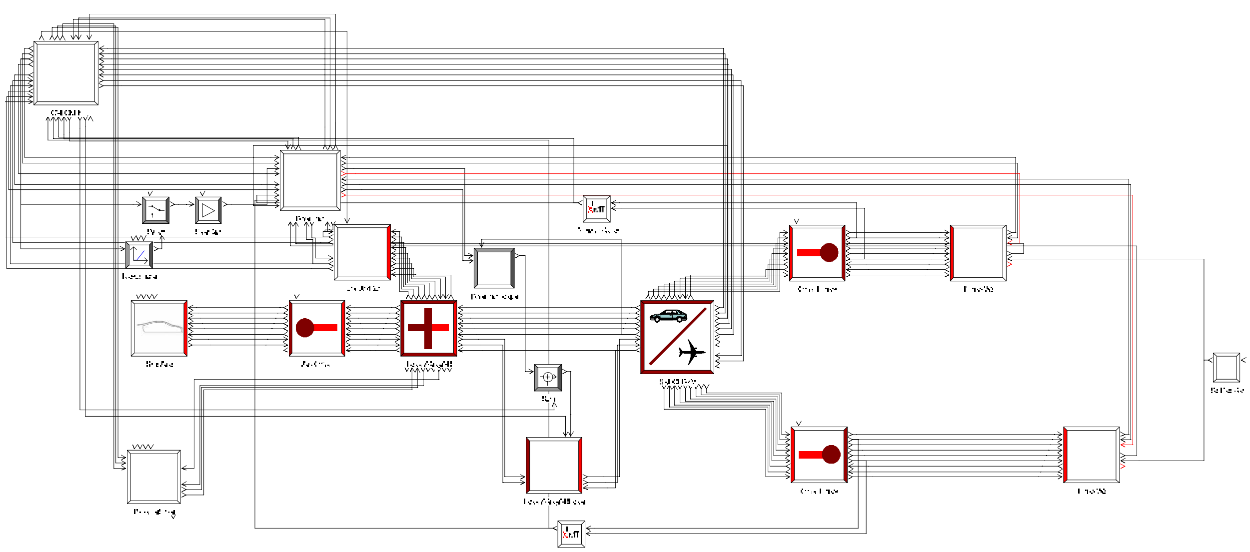

Figure 14. Top-level vehicle dynamics model as implemented in SimCreator.

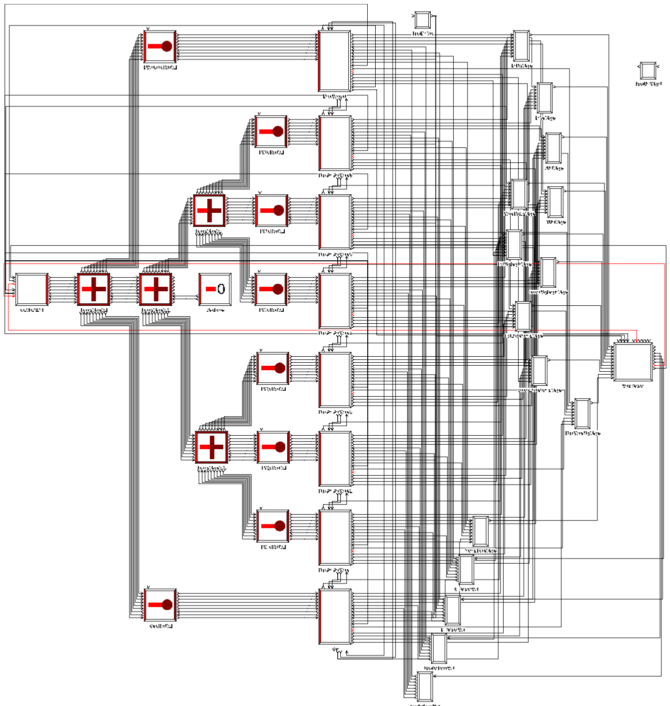

Figure 15. Track model for six road wheels as implemented in SimCreator.

motion-base and P&E SIL's power system hardware to interact together as if they were both connected locally. Both the TARDEC and P&E SIL contain coupled dynamic systems that create a seamless simulation environment for realistically exercising the power train hardware located in Santa Clara, CA. Remote operation of the P&E SIL hardware is initiated by a human operator in a driving simulation environment located at the TSL in Warren, MI where a vehicle dynamics model is simulated locally to drive a motion base simulator. These two test sites are separated by 2,450 miles (see Figure 16) but communicate over the open Internet. Use of the open Internet as a communication channel to couple these two dynamic systems poses several problems [5] including significant time delay, variable time delay, and data loss.

The initial design of the *RemoteLink* consisted of the four strategies shown below.

1. Local power system model: A dynamic model of the entire SIL power system, CHPSPerf [11,12], running on the crewstation motion base. This model provides an estimate of the real power system hardware's response to the motion base vehicle model.
2. Adaptive filtering algorithm: A Kalman or recursive least squares (RLS) filter to provide real-time updates to the TSL mobility model's torque inputs [14].
3. State convergence: A method for observing and coordinating pertinent dynamic states for both the mobility and power system models implemented at both ends of the connection.
4. Parameter tuning: Future work includes both offline and online parameter estimation for the power system model. CHPSPerf is validated against experimental data, however, both extended hardware operation and temperature-dependent effects present a need for continued power system parameter estimation.

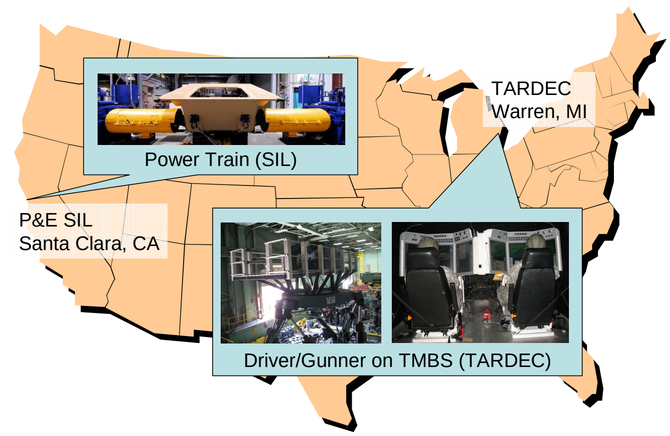

Figure 16. Locations of the DCE3 simulator assets

The inclusion of the local power system model in the TSL was necessary to enable the driver to receive instantaneous response to his/her inputs. Without the presence of the local power system model, the TSL vehicle would have responded on the order of two one-way communication delays, which could have made the vehicle un-drivable, particularly for steer.

The inclusion of the state convergence algorithms was another crucial piece of the RemoteLink design. *State convergence* algorithms are observers which cause the states in the TARDEC and P&E SIL to track each other. Specifically, the power system model states in the TSL must track the states of the P&E SIL power system hardware. This observer is called the *powertrain observer*. Similarly, the states in the P&E SIL vehicle model should track the states of the TSL vehicle model. This observer is called the *vehicle dynamics observer*. Without the presence of the state convergence algorithms, the P&E SIL and TSL vehicle models would diverge in position to different locations on the Ft. Knox course, which would render the experiment meaningless.

Figure 17 shows the relationships between the RemoteLink strategies in detail as well as how all of the relevant components of the TSL and P&E SIL are connected together. The TSL (shown in the top half of Figure 17) consists of three main features. The first is the driver, who operates a crewstation mounted on a motion simulator. The crewstation receives both visual and motion feedback which provides the driver with a realistic driving experience and the P&E SIL power system with realistic driver commands. To provide feedback to the driver, the crewstation and motion simulator both use vehicle states from the local vehicle model, which is the second important feature of the TSL. This vehicle model receives sprocket torques from the power system model and sends vehicle states to the crewstation and motion simulator. The third TSL feature

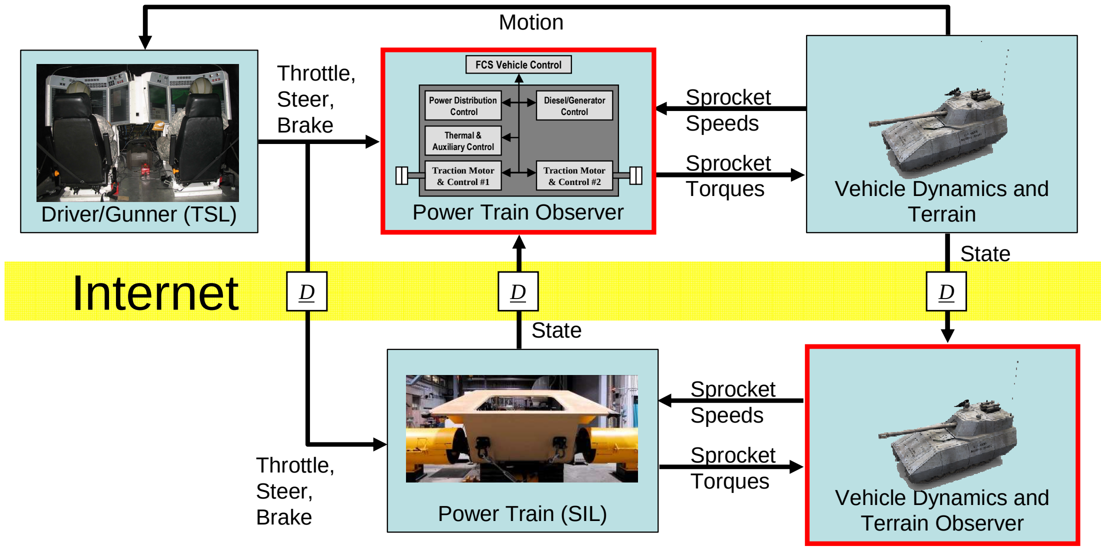

Figure 17. RemoteLink Architecture.

is the CHPSPerf hybrid electric power system model [12]. The CHPSPerf power system is designed to closely model the hardware in the SIL and provides high frequency torque response to the TARDEC vehicle model. These torques are computed based upon driver commands, vehicle states, local motor models, and torque data from the P&E SIL. The local CHPSPerf power system model has two objectives: 1) provide fast, realistic response to the driver to maintain realistic driver feel and 2) provide a response that closely resembles the behavior of the actual P&E SIL power system. Note that the power system torques are not physical torques, but are torques that exist in software. With respect to the P&E SIL (shown in the bottom half of Figure 17), two major entities are present – the series hybrid power system hardware designed to power a 20-22 ton tracked vehicle and the vehicle model and dynamometers. The power system hardware receives a time delayed version of the driver inputs *(steering, throttle, and braking)* from the TSL along with the vehicle speed from the SIL vehicle model and responds with actual traction motor torques. The vehicle model computes speed states and produces reaction torque commands which result from interaction with virtual terrain. These load torque commands are fed back to the power system through dynamometers that are connected to the traction motors. This load emulation process is described in more detail in [4]. On the SIL side, the *state convergence* algorithms reside in the vehicle model and should cause its states to track those of the vehicle model at TARDEC.

SYSTEM OPERATION

The system operation of the *RemoteLink* is initiated by driver inputs. Once the driver provides inputs (steering, throttle, and braking) to the crewstation, those inputs flow simultaneously to TARDEC and to the P&E SIL. However, the driver inputs must travel through the open Internet and across the country in order to reach the P&E SIL. Thus, the TSL power system model receives driver inputs before the SIL power system hardware receives those same driver inputs. For reference purposes, suppose that the driver supplies commands at time t and that the one-way cross country Internet delay is a constant value of Δ. This implies that the driver won't feel the response from the SIL hardware until time $t+2\Delta$. If Δ is too large, the driver won't be able to navigate the vehicle in a stable fashion. This illustrates the importance of having the local power system model at TARDEC to provide an instantaneous response.

The downside to having a local power system model is that a model of a system can never perfectly match the physical system. Thus, the presence of the TARDEC power system model introduces error between the TARDEC torques and the P&E SIL torques. Therefore, one can deduce that if the torques are in error, then other vehicle states such as sprocket speeds, velocity, and positions will be in error. Once the TSL states become significantly different from the P&E SIL states, the driver-in-the-loop/hardware-in-the-loop experiment loses meaning. Avoiding divergent states is the motivation for state convergence within the RemoteLink. The initial investigations of the RemoteLink led us to examine two methods for achieving state convergence, in the end, however, the method based on sliding mode control was used as described next (the second method is described in [15]).

SLIDING MODE CONTROL APPROACH

The state convergence method used a robust control algorithm called Sliding Mode Control. A derivation of Sliding Mode Control is outside the scope of this paper (see Slotine & Li [10] for a full development); however, the application of Sliding Mode Control for *state convergence* is presented below. Before showing the application of Sliding Mode control to the *RemoteLink*, first it is necessary to make a few general remarks about sliding mode control.

Sliding Mode Control can transform a higher order tracking problem into a first-order stabilization problem [10]. The main idea of sliding mode control is to drive the states of the system to a desired area in state-space known as the sliding surface, which is defined by the designer. Let us assume that the system we are modeling has the form of

$$\begin{aligned} \dot{x} &= f(x) + b(x,t)u(t) \\ y &= h(x) \end{aligned} \tag{1}$$

where u is the control input, x is the state vector, and y the output vector.

In addition, suppose that the vector y represents the actual outputs, or states, of a system and the vector y_d represents the desired system outputs. A commonly used sliding surface [10] is

$$s(x,t) = (\frac{d}{dt} + \lambda)^{(n-1)}\, \tilde{y}(t) \tag{2}$$

where n is the relative order of the output, $\tilde{y}(t)$ is the output error $y(t) - y_d(t)$, and λ is a constant chosen by the designer.

In effect, this sliding surface, s, is an error surface. It is desirable to maintain this error surface at zero; hence, it is shown that this tracking problem is transformed into a first order stabilization problem in s. Not only is it desirable to maintain the error at zero, but it is also desirable to maintain the error rate-of-change at zero. Due to the fact that the order of (2) is n-1, s only needs to be differentiated once for the input to appear. Taking the derivative of s and setting it equal to zero leads to a solution for the equivalent control term.

The equivalent term is an important and necessary component of sliding mode control, but an additional term is needed to maintain the sliding mode in the presence of disturbances, modeling simplifications and parametric uncertainties. This term is called the robust term. The robust term [10] typically takes the form $u_{rob} = -\eta\,\mathrm{sgn}(s)$, however the *state convergence* implementation is modified slightly to use the $\tanh()$ function,

$$u_{rob} = -\eta \tanh(s / s_0) \tag{3}$$

where η is a constant chosen by the designer and s_0 is a boundary layer width. Unlike the $\mathrm{sgn}()$ function the $\tanh()$ function is smooth near zero and with a properly sized boundary layer, eliminates chatter near $s = 0$.

Therefore, the robust term works by aggressively forcing the system back to the sliding mode when the states leave the boundary layer around the original sliding surface $s = 0$.

Now that a generalized method for deriving an effective sliding mode control law has been provided, this method can be applied to the *RemoteLink*. In the case of the *RemoteLink*, the goal is to make the states of the P&E SIL vehicle model follow the states of the TARDEC vehicle model. Two approaches can be taken with respect to *state convergence* of the vehicle states: 1) force the P&E SIL states to track the TSL states as quickly and abruptly as possible 2) gradually migrate the P&E SIL states to track the TSL states. In the interest of protecting the P&E SIL power system hardware, the second approach of gradually nudging the P&E SIL states is chosen.

The aforementioned states that must be converged include global X position, global Y position, velocity, left sprocket speed, right sprocket speed, and yaw angle. In the interest of being brief, the derivation is only shown for yaw angle *state convergence*. Derivations for the other states are similar to the following derivation for yaw angle *state convergence*.

The first step in deriving a sliding mode control law is obtaining the equivalent control term, and the first step in obtaining the equivalent control term is to write the equation of motion for the state of interest. Thus, the equation of motion for the vehicle yaw angle, Ψ, is

$$\ddot{\psi} = \frac{\Sigma M_z - (\bar{\omega} \times \bar{J} \cdot \bar{\omega})_z}{J_{zz}} + p_\psi \tag{4}$$

where M_z is the moment about the yaw axis, ω is the angular velocity of the vehicle, J is the vehicle rotational moment of inertia, and p_ψ is the *state convergence* control input.

As indicated in the above methodology for sliding mode control, the next step is to define a sliding surface for the control to follow. Accordingly, a sliding surface is defined as

$$s = \dot{\tilde{\psi}} + \lambda\tilde{\psi} = \dot{\psi}_{SIL} - \dot{\psi}_{TAR} + \lambda(\psi_{SIL} - \psi_{TAR})\,. \tag{5}$$

Taking the time derivative of the sliding surface and setting the equation equal to zero reveals the following expression.

$$0 = \ddot{\psi}_{SIL} - \ddot{\psi}_{TAR} + \lambda(\dot{\psi}_{SIL} - \dot{\psi}_{TAR}) \tag{6}$$

Examining (6), we see that terms exist for the P&E SIL and for TARDEC. Note that the TARDEC yaw rate and yaw acceleration terms are desired values coming across the network from TARDEC to the P&E SIL. To bring the control input term into this equation, the yaw acceleration defined in (4) must be substituted into (6). After re-arranging terms, the expression for the equivalent control term is

$$p_{\psi,eq} = \ddot{\psi}_{TAR} - \lambda(\dot{\psi}_{SIL} - \dot{\psi}_{TAR}) - \frac{\Sigma M_z - (\bar{\omega} \times \bar{J} \cdot \bar{\omega})_z}{J_{zz}} . \quad (7)$$

As mentioned above, the TARDEC terms are available from information coming over the network connection. The P&E SIL terms are all accessible from the vehicle dynamics model in the P&E SIL. The equivalent control term is a necessary component of sliding mode control, but it alone is not enough to guarantee robust controller performance.

To withstand disturbances or modeling uncertainty, a second (robust) term is necessary, as defined in (3) above. The designer must choose the gain parameter η at a level large enough such that the controller has enough authority to drive the states to within their boundary layers. However, a gain parameter which is too large may cause numerical instability, chatter, and unmodeled high-frequency system dynamics. The robust term is simply added with the equivalent term to get the complete non-linear sliding mode control

$$\begin{aligned} p_{\psi} = {} & \ddot{\psi}_{TAR} - \lambda(\dot{\psi}_{SIL} - \dot{\psi}_{TAR}) \\ & - \frac{\Sigma M_z - (\bar{\omega} \times \bar{J} \cdot \bar{\omega})_z}{J_{zz}} - \eta \tanh(s / s_0) \end{aligned} . \quad (8)$$

The sliding mode control input shown in (8) is implemented into the tracked vehicle dynamics model [12] in the Matlab/Simulink simulation environment. Similar derivations are performed and implemented in simulation for the other five states that must be converged.

RESULTS

DEMOGRAPHICS

The DCE3 experiment included military subjects from the Future Forces Integration Directorate (FFID) Operations from Ft. Bliss, Texas. All subjects were male and the average age was 27 years old. All of the subjects had reported either "none" or "mild" Simulator Sickness or Motion Sickness in the past. Their educational background included at least high school level for 11 out of 13 subjects, with two subjects completing a Bachelors Degree. The military data for these subjects included two commissioned officers and 11 NCOs (Non-Commissioned Officer). All of the military subjects had an MOS (Military Occupational Specialty) of 19K (M1 armor crewman). However, one subject had an MOS of 19A (commissioned officer as an armor crewman). There were four gunners, three commanders, two platoon leaders, three platoon sergeants and one loader/driver. There was an average of 78 months of service time between the 13 military subjects. All of them had experience in an M1A1/A2/A2SEP ground vehicle and ten of them had additional experience in a HMMWV, and three had additional experience in an M1113. Nearly half (7) of them have computer gaming experience and have spent time in Iraq or Afghanistan.

DUTY CYCLES

Of the twelve teams which performed the UAMBL scenarios, eleven of them ran to completion, the other had to be aborted mid-way through because the vehicle rolled over (which was resumed from the point of rollover). Of the twelve teams ten of them ran the CASTFOREM scenario. Of these ten, seven ran to completion, the other three had to be stopped for technical reasons; they were resumed from the point of stoppage and run to completion. Although designed to execute for each of the runs, the P&E SIL was only able to operate in one of the scheduled runs.

Regarding the actual duty cycles, all pertinent vehicle and power system data were recorded for each run and archived for further use and analysis. All crew behaviors were recorded to include instantaneous driver and gunner commands. For those runs with which the SIL ran, time-correlated SIL data were recorded. For non-mobility loads all of the fire and detonation events for both the red and blue forces were logged.

As an example of the types of data that were recorded, Figure 18 shows the paths of all twelve teams through

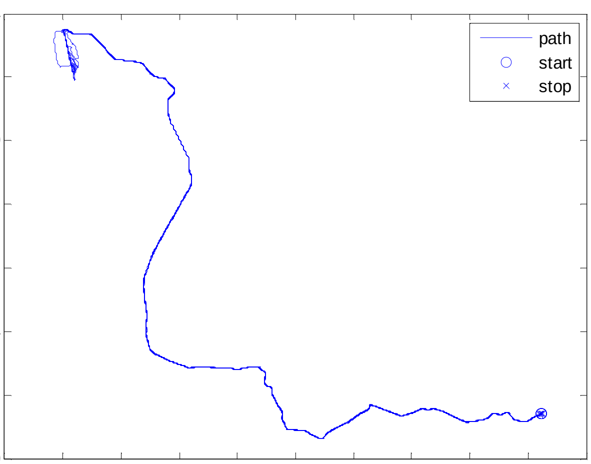

Figure 18. Overlaid path of all twelve experiment runs over all 61 km of the scenario. (Not to scale)

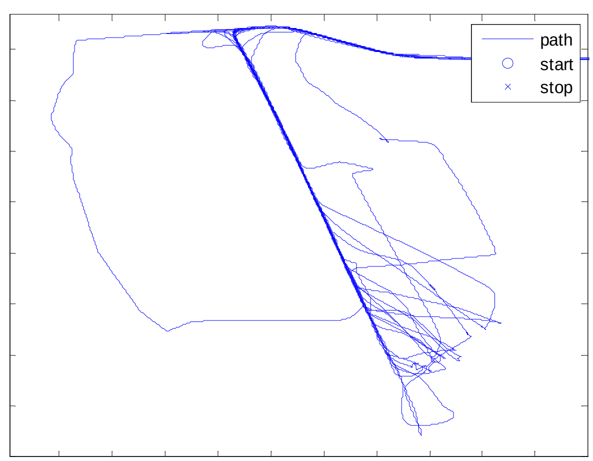

Figure 19. Close-up of the overlaid paths during the tactical maneuver portion of the scenario. (Not to scale)

the whole UAMBL scenario. Observe that there is consistency while the vehicles are on route IRISH. After the operators reach the SP, they were free to maneuver tactically to set the support by fire position observed in the upper-left of the figure. Figure 19 shows a close-up of the paths taken in the SBF portion of the scenario. In this case, it can be seen that most crews took a particular road south to the SBF position. Two crews over shot the turn. Of these one crew returned back to the main path; the other crew took a different route and set a different SBF position.

The definition of a duty cycle also includes the events and circumstances associated with each point on the path driven. Because each team negotiated the course at different speeds, plots with time as the independent variable introduce skew among events. For this reason the following plots are shown as functions of distance along the course. First we examine the terrain features along the route as shown in Figure 20. There we observe the rich variety of elevation and grades encountered by the vehicle along the route. Also included in the definition of a duty cycle are the behaviors of the crew along the route. First we observe the longitudinal commands of the driver in Figure 21 and of the lateral performance of the driver in Figure 22. Next, the duty cycle definition may also include the activity of the turret and gun as illustrated in Figure 23 and particular vehicle components as illustrated with the major SIL components in Figure 24.

CONCLUSION

In this paper we described a human-in-the-loop motion-based simulator which was designed, built and used to measure the duty cycle of a combat vehicle in a virtual simulation environment. The simulation environment integrated two advanced crewstations for a driver's station and a gunner's station of a simulated future tank. The simulated systems of the tank include a series hybrid-electric propulsion system and its main weapon systems. The vehicle was placed in two different virtual combat scenarios which were then executed by the participating Soldiers. The duty cycle was measured as the commands of the driver and gunner as well as terrain and enemy contact. We discussed the motivation and approach to integrating the P&E SIL in real-time with the simulation at TARDEC. Such an integration used the Internet as a communication channel and the design accounted for the unreliable nature of the channel.

After having successfully completed the DCE1, DCE2 and DCE3 experiments TARDEC has planned an additional three follow-on experiments in FY08. These experiments will subsequently be called DCE4, DCE5, and DCE6. They will perform the same evaluations and measurements for future tactical vehicles (i.e. trucks). These will be executed in scenarios which are markedly different than those for DCE1 – DCE3. These scenarios will be designed for tactical vehicle type missions.

ACKNOWLEDGMENTS

We the authors of this paper wish to acknowledge Mr. Patrick Nuñez who was taken from us on June 19, 2007. It was Pat who had the idea of integrating the motion base simulator, the CAT crewstations and the P&E SIL for the purpose of measuring combat vehicle duty cycles. Without Pat's vision and initiative the tremendous advances described in this paper would not have happened. Like all of his ideas, it was "bold."

REFERENCES

1. Brudnak, M.; Nunez, P.; Paul, V.; Mortsfield, T.; Shvartsman, A.; Perera, H.S.; Smith, W.; Mohammad, S.; Pozolo, M., "Human-in-the-loop Simulation-based Combat Vehicle Duty Cycle Measurement: Duty Cycle Experiment 1", Paper SIW-06S-080, SISO, Orlando, FL, April 2006.
2. Brudnak, M.; Pozolo, M.; Paul, V.; Mohammad, S.; Smith, W.; Compere, M.; Goodell, J.; Holtz, D.; Mortsfield, T.; Shvartsman, A., "Soldier/Hardware-in-the-loop Simulation-based Combat Vehicle Duty Cycle Measurement: Duty Cycle Experiment 2," Paper SIW-07S-016, SISO, Norfolk, VA, March 2007.
3. Compere, M.; Goodell, J.; Simon, M; Smith, W.; Brudnak, M, "Robust Control Techniques Enabling Duty Cycle Experiments Utilizing a 6-DOF Crewstation Motion Base, a Full Scale Combat Hybrid Electric Power System, and Long Distance Internet Communications", Paper 2006-01-3077, SAE Power Systems Conference, Nov. 2006.
4. Compere, M.; Simon, M.; Kajs, J.; Pozolo, M., "Tracked Vehicle Mobility Load Emulation for a Combat Hybrid Electric Power System", Paper presented at the 6th International AECV Conference, June 2005.
5. Elhajj, I.; Xi, N.; Fung, W.K.; Liu, Y.H.; Hasegawa, Y.; Fukuda, T.; "Supermedia Enhanced Internet Based Telerobotics", in Proceedings of the IEEE, special issue on Networked Intelligent Robots Through the

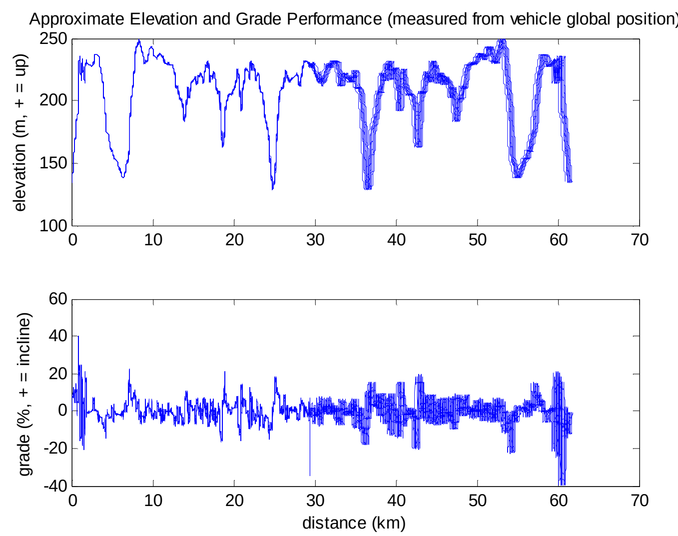

Figure 20. Overlaid plot of terrain for all twelve runs as a function of distance. Included are the elevation (top) and grade (bottom).

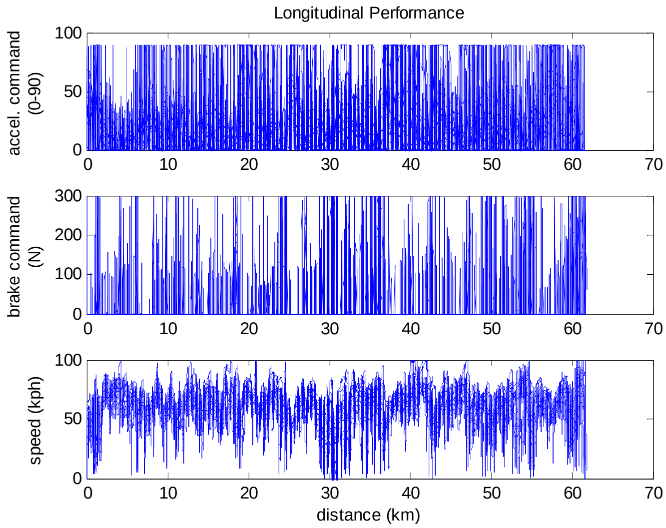

Figure 21. Overlaid plot of the vehicle's longitudinal performance for all twelve runs as a function of distance. Included are the throttle (top), the brake (middle) and speed (bottom).

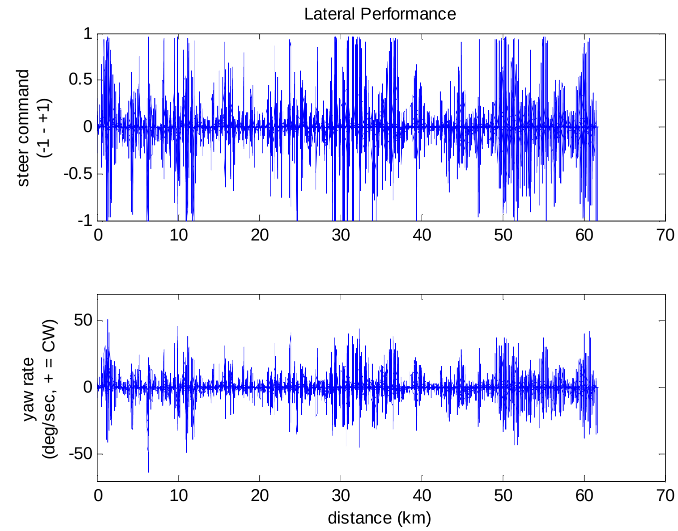

Figure 22. Overlaid plot of the vehicle's lateral performance for all twelve runs as a function of time. Included are the steer (top) and yaw rate (bottom).

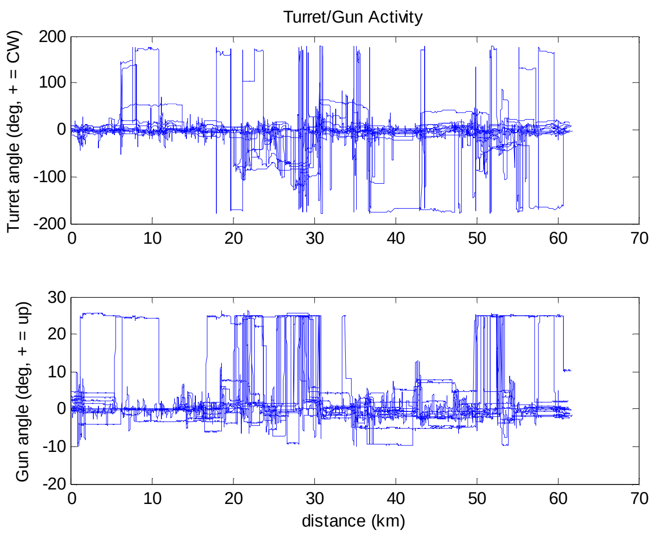

Figure 23. Overlaid plot of the vehicle's weapon system performance for all twelve runs as a function of time. Included are the turret azimuth (top) and gun elevation (bottom).

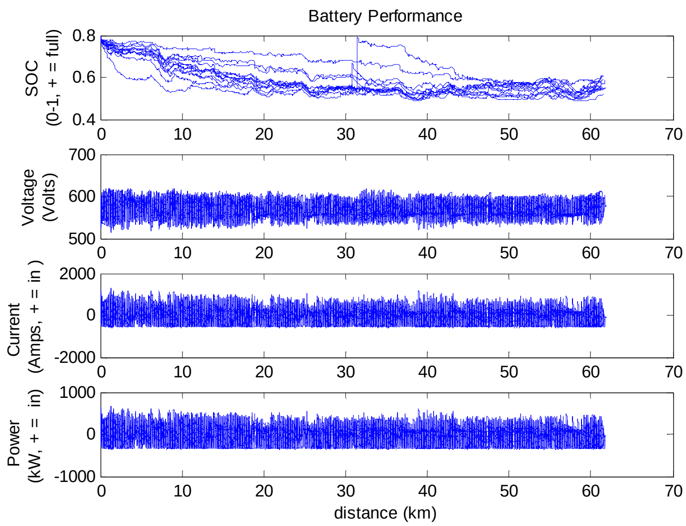

Figure 24. Overlaid plot of the vehicle's battery performance. (from the top) the state of charge, battery Voltage, battery current, battery power.

Internet, Vol. 91, No. 3, pp. 396-421, March 2003.

6. Goodell, J., Compere, M., Simon, M., Smith, W., Brudnak, M., Pozolo, M., Paul, V., Mohammad, S., Holtz, D., Mortsfield, T., Shvartsman, A., "Internet Enabled Remote Driving of a Combat Hybrid Electric Power System for Duty Cycle Measurement", 7th International All Electric Combat Vehicle Conference, Stockholm, Sweden, June 11-13, 2007.
7. Paul, V.; Brudnak, M.; Ueda, J.; Shvartsman, A., "Simulation-based Hybrid Electric Combat Vehicle Duty Cycle Measurement", Intelligent Vehicle Systems Symposium, NDIA, Traverse City, MI, June 2006.
8. Romano, R., "Realtime Driving Simulation Using a Modular Modeling Methodology", Paper 2000-01-1297,SAE, March 2000.
9. Romano, R. "Real-Time Multi-Body Vehicle Dynamics Using a Modular Modeling Methodology", Paper 2003-01-1286, SAE, March, 2003.
10. Slotine, J.-J., Li, W, *Applied Nonlinear Control*, Prentice Hall, Englewood Cliffs, NJ, 1991.
11. Smith, W.; Nunez, P., "Power and Energy Computational Models for the Design and Simulation of Hybrid-Electric Combat Vehicles", SPIE Paper No. 5805-02, Paper Presented at Simulation Science IX, Orlando, FL, 28 March – 1 April 2005.
12. Smith, W.; and Wehmueller, K., "Detailed Simulation and Development of Hybrid Power Systems Energy Management Architectures", Paper presented at the 4th International AECV Conference, January 2002.
13. Wong, J.Y. *Theory of Ground Vehicles*, 3rd Ed., John Wiley & Sons, ISBN 0-471-35461-9, 2001.

14. Wright, R.; Goodell, J.; Smith, W.; Nunez, P., “A Delay Compensation Strategy for Improved Remote Link Control Over A Long Haul Communications Channel”, Paper presented at the 6th International AECV Conference, June 2005.
15. Goodell, J; Compere, M; Simon, M; Smith, W; Wright, R; Brudnak, M, "Robust Control Techniques for State Tracking in the Presence of Variable Time Delays", Paper 2006-01-1163, SAE, April 2006

CONTACT

Dr. Mark Brudnak may be contacted at:

U.S. ARMY TARDEC
ATTN: AMSRD-TAR-R/157
6501 E. 11 Mile Rd.
Warren, MI 48397-5000

ACRONYMS

ATO: Army Technology Objective.
BLOS: Beyond Line of Sight (engagement).
CASTFOREM: Combined Arms and Support Task Force Evaluation Model.
CAT: Crew-integration and Automation Test-bed.
CERDEC: (Army) Communications and Electronics Research Development and Engineering Center.
CHPS: Combat Hybrid Power System.
CHPSPerf: Combat Hybrid Power System Performance (Modeling tool).
CS/TMBS: Crew Station / Turret Motion Base Simulator.
DCE: Duty Cycle Experiment.
DIS: Distributed Interactive Simulation.
DOF: Degree of Freedom.
DSS: Decision Support System.
ESS: Embedded Simulation System.
FCNet: Fire Control Network.
FCS: Future Combat System.
GVW: Gross Vehicle Weight.
HMMWV: High Mobility Multi-purpose Wheeled Vehicle.
HWIL: Hardware-in-the-loop.
ICV: (FCS) Infantry Carrier Vehicle.
IED: Improvised Explosive Device.
IG: Image Generator.
IR: Infrared (sight).
LOS: Line of Sight (engagement).
MC2: Mobile Command and Control.
MFD: Multi-Functional Display.
MGV: (FCS) Manned Ground Vehicle.
MOS: Military Occupational Specialty.
NCO: Non-Commissioned Officer.
NV: Night Vision.
NVIG: Night Vision Image Generator.
NVL: Night Vision Laboratory.
OE: Operating Environment.
OneSAF: One Semi-Automated Forces.
OTB: OneSAF Test Bed.
P&E: Power and Energy.
RDECOM: (Army) Research Development and Engineering Command.
RMS: Ride Motion Simulator.
RPG: Rocket Propelled Grenade.
RSTA: Reconnaissance Surveillance and Target Acquisition.
RLS: Recursive Least Squares.
RTI: Realtime Technologies, Inc.
RTW: (Simulink) Real-Time Workshop.
SBF: Support By Fire.
SFC: Specific Fuel Consumption.
SIL: System Integration Laboratory.
SMI: Soldier Machine Interface.
SOC: State of Charge.
TARDEC: Tank Automotive Research, Development, and Engineering Center.
TMBS: Turret Motion Base Simulator.
TSL: TARDEC Simulation Laboratory.
UAMBL: Unit of Action Maneuver Battle Laboratory.
UGV: Unmanned Ground Vehicle.
VB-IED: Vehicle Bourne - IED.
VTT: Vetronics Technology Testbed..